ＭＯＮＰＥ−１．０．０
サンプルプログラム

ソフトウェア購入申請書
取扱説明書

平成１６年２月

（株）ネットワーク応用通信研究所

目次

1.準備……3

システム概要……3
動作環境……4
インストール方法……5
起動……7
注意事項……7

2.ログイン……8

Step1.アカウント情報の入力……8
Step2.ログイン……9

3.処理選択画面の概要……10

4.新規作成……11

Step1.新規作成の選択……11
Step2.必要な項目の入力……11
Step3.入力内容の確認……11
Step4.作成完了……14
Step5.印刷……15

5.閲覧……16

Step1.閲覧の選択……16
Step2.閲覧するソフトウェア購入申請書の選択……16
Step3.ソフトウェア購入申請書の閲覧……17

6.修正……18

Step1.修正内容の入力……18
Step2.修正内容の確認……19
Step3.修正完了……21
Step4.印刷……21

7.削除……22

Step1.削除の確認……22
Step2.削除完了……23

8.印刷……24

Step1.印刷の実行……24

9.ログアウト……25

Step1.ログアウトの選択……25
Step2.本システムの終了……25

付録 A.ディレクトリ構成……26

付録 B.ユーザの追加・修正・削除 SQL……27

付録 C.プリンタの追加・修正・削除 SQL……28

1.準備

ここでは、本システムの概要やインストール方法などを説明します。

システム概要

本システムは、図 1.1 に示すソフトウェア購入申請書を印刷するWeb ベースのシステムです。
印刷に必要な情報はデータベースで管理し、内容の修正や削除といったことができます。

平成 16年 4月 1日

ソフトウェア購入申請書

承認	上司	担当者

氏名	プロジェクト名	プロジェクトコード
TEST1 太郎	プロジェクト1	10000

購入理由

オープンソースソフトウェアで業務システムを構築するため

予定時期	購入予定ソフトウェア	購入予定先	個数	購入予定金額
4/1	MONTSUQI	NaCl	1	10,000
5/1	MONPE	NaCl	1	10,000
6/1	MONSIA	NaCl	3	30,000
		合 計	5	50,000

図 1.1 ソフトウェア購入申請書

本システムを利用するユーザ(クライアント)は、Web ブラウザを利用して本システムを操作します。クライアントからの要求は、サーバに送られ、サーバ上で合計金額の計算などの処理やデータベースアクセスなどをおこない、その結果をユーザに返します。図 1.2 は、本システムの構成概要図です。

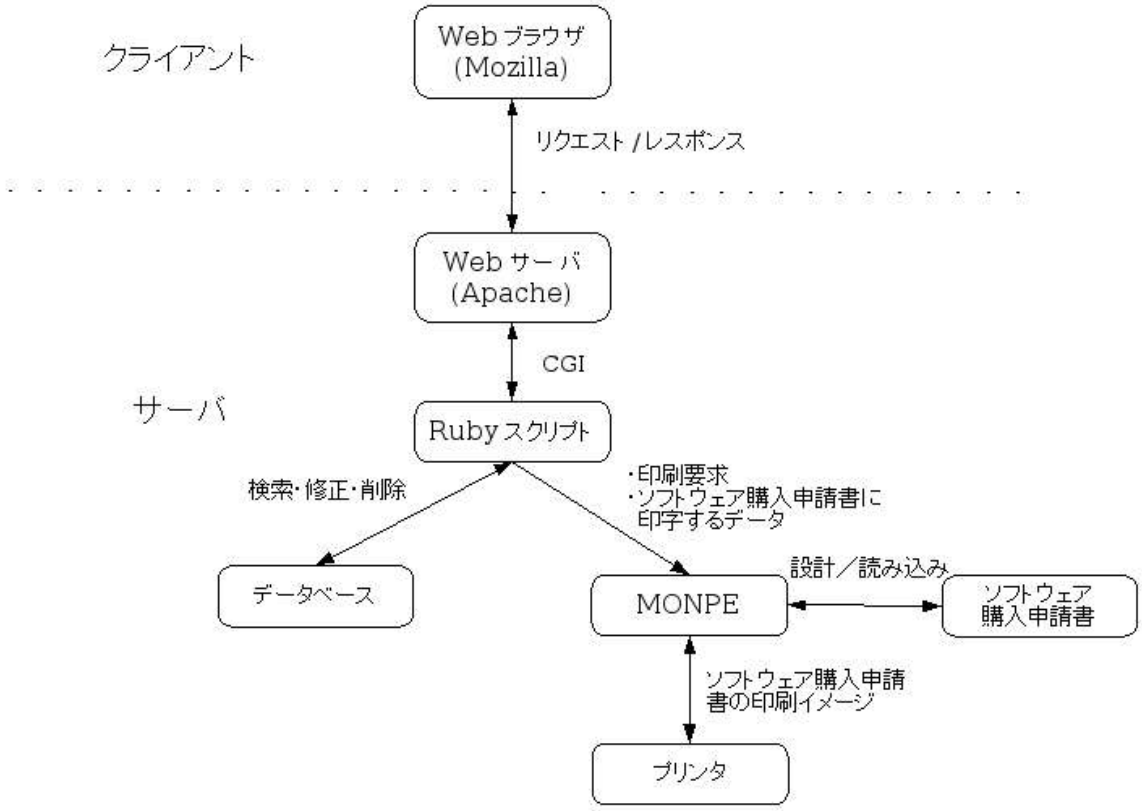

図 1.2 システム構成概要図

動作環境

開発および動作確認をおこなったのは次のような環境です。

サーバ側

ハードウェア

CPU:Pentium III 500MHz

メモリ:320MB

HDD:40GB

プリンタ: FUJI XEROX DocuCentre Color 320 CP

ソフトウェア

OS:Linux 2.4.18

Linux ディストリビューション: Debian GNU/Linux 3.0(Woody)

Web サーバ: Apache 1.3.26

DBMS: PostgreSQL 7.2.1

Ruby 1.6.7

eruby 0.9.2

mod_ruby 0.9.7

ruby-postgres 0.6.5

MONPE 1.0.0

LPRng 3.8.10

クライアント側

Web ブラウザ：Mozilla 1.0

以降では、Debian GNU/Linux 3.0(Woody)で本システムを動作させると仮定して説明します。

インストール方法

本システムをインストールするには、次のことをおこなう必要があります。また、インストールはユーザ kouji でおこなうと仮定します。

1. プログラムのインストール
2. Web サーバ(Apache)の設定
3. DBMS の設定
4. データベースの構築
5. プリンタの設定

それでは、順番に説明していきます。

プログラムのインストール

本システムのソースコードは、MONPE のプロジェクトページから取得できます。

ユーザ kouji で次のコマンドを実行し、softshinsei.tar.gz を~/public_html 以下に取得、解凍します。

これでプロログラムのインストールは完了です。

```
$ mkdir -p ~/public_html
$ cd ~/public_html
$ wget ftp://ftp.montsuqi.org/pub/archive/monpe/softshinsei.tar.gz
$ tar xvzf softshinsei.tar.gz
```

Web サーバ(Apache)の設定

本システムを動作させるには、Web サーバ上で Ruby スクリプトが動作しなければいけません。

しかし、標準の設定では、そのようになっていません。そこで、Web サーバである Apache の設定をして、Web サーバ上で Ruby スクリプトが動作するようにします。

Apache の設定ファイルである/etc/apache/httpd.conf を次のように変更します。

- 243 行目付近のコメントを外す。

```
LoadModule ruby_module /usr/lib/apache/1.3/mod_ruby.so
```

- 383 行目付近に ExecCGI を追加する。

```
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec ExecCGI
```

- 781 行目付近のコメントを外す。

```
    AddHandler cgi-script .cgi .sh .pl
```

- ファイルの最後に以下を追加

```
<IfModule mod_ruby.c>
  RubyRequire apache/ruby-run
  RubyRequire apache/eruby-run
  RubyPassEnv LANG
</IfModule>
```

変更後、Apache を再起動すれば、Ruby スクリプトが動作するようになっています。

これで、Web サーバの準備が整いました。

DBMS の設定

DBMS のユーザ認証について記入するファイル「/etc/postgresql/pg_hba.conf」に、次のように記述し、DBMS を再起動してください。

```
local        all                                              trust
host         all         127.0.0.1      255.255.255.255     trust
host         all         0.0.0.0        0.0.0.0             crypt
```

データベースの構築

続いて、本システムに必要なデータベースを構築します。

まず、ユーザ kouji にデータベースの作成権限を与え、report という名前のデータベースを作成します。

```
$ su -
<root ユーザのパスワード>
# su - postgres
$ createuser kouji
Shall the new user be allowed to create databases? (y/n) y
Shall the new user be allowed to create more new users? (y/n) y
$ exit
# exit
$ createdb report
```

そして、本システムのソースコードに含まれる sql/insert-test.sh を実行します。

```
$ cd ~/public_html/softshinsei/sql
$ ./insert-test.sh
```

これで、本システムに必要なデータベースが構築できました。

プリンタの設定

最後は、プリンタの設定です。本システムで使用する lpr エントリ名「lp」を/etc/printcap の lp を適宜修正します。以下には LPRng インストール時の lp エントリ名の情報を示します。

```
lp|Generic dot-matrix printer entry:¥
        :lp=/dev/lp0:¥
        :sd=/var/spool/lpd/lp:¥
        :af=/var/log/lp-acct:¥
        :lf=/var/log/lp-errs:¥
        :pl#66:¥
        :pw#80:¥
        :pc#150:¥
        :mx#0:¥
        :sh:
```

これで、プリンタの設定ができました。

起動

それでは、Web ブラウザを使って、本システムの動作確認をおこないましょう。

Web ブラウザで、http://localhost/~kouji/softshinsei/login.cgi にアクセスしてみてください。

図 1.3 のようなログイン画面が表示されれば、本システムが正常に動作しています。

図 1.3 起動画面

注意事項

本システムを利用するには、次の情報をあらかじめをデータベースに登録しておく必要がありますが、本システムでは、これらの情報の追加、更新、削除といった処理ができません。

- **ユーザ名やパスワードなどのユーザ情報**
- **/etc/printcap に設定したプリンタ名やその説明などのプリンタ情報**

しかし、データベースを構築した際に本システムが動作するための最低限の情報は登録されていますので、動作確認には、これらの情報を使ってください。

もし、ユーザ情報、プリンタ情報の追加、更新、削除がおこなう場合は、直接 SQL を発行する必要があります。それぞれをおこなう SQL は、付録 B、C で説明していますので、こちらを参照ください。

以降では、データベースを構築した際に登録された情報を用いて、本システムを使用すると仮定して説明します。

2.ログイン

本システムを使用するには、ユーザの指定とパスワードの入力をおこない本システムにログインする必要があります。

ここでは、本システムにログインする方法について説明します。

Step1.アカウント情報の入力

本システムにアクセスすると、まずは図 2.1 のようなログイン画面が表示されます。

ユーザ選択リストから、ログインするユーザを選択し、パスワード入力欄にパスワードを入力し、ログインボタンを押してください。

図 2.1 ログイン画面

表 2.1 にテストユーザのユーザ名とパスワードを示します。

ユーザ名	パスワード
TEST1　太郎	test1
TEST2　二郎	test2
TEST3　花子	test3
TEST4　桃子	test4

表 2.1 テストユーザのアカウント情報

Step2.ログイン

パスワードが正しければ、図 2.2 のようなログイン中画面が数秒表示され、その後、自動的に処理選択画面が表示されます。これで、ログインができましたので、ソフトウェア購入申請書の作成や印刷といった処理をおこなうことができます。

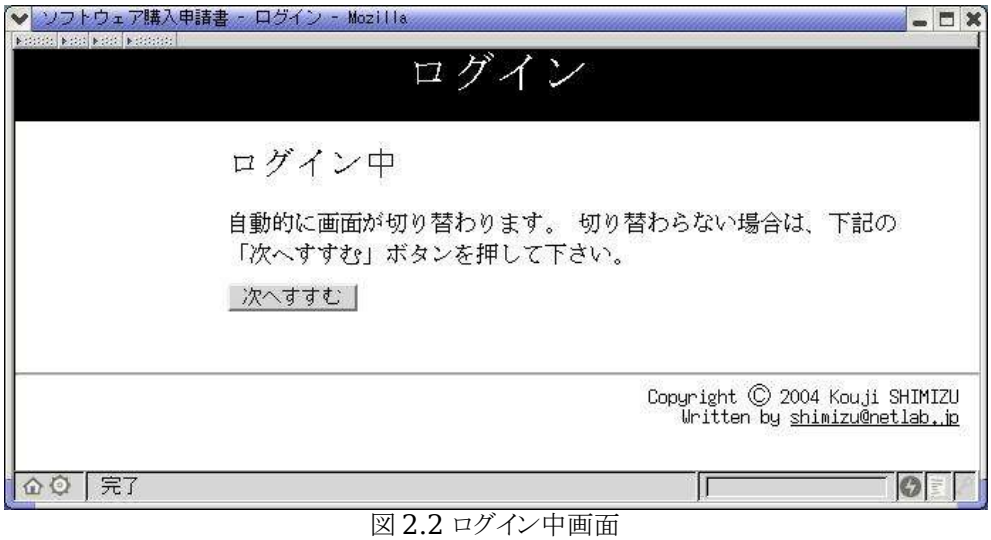

図 2.2 ログイン中画面

パスワードが正しくなければ、ログインエラー画面が表示されますので、その場合は、再度ログインしてください。

3.処理選択画面の概要

ここでは、処理選択画面について説明します。
ログインすると、図 3.1 のような処理選択画面が表示されます。

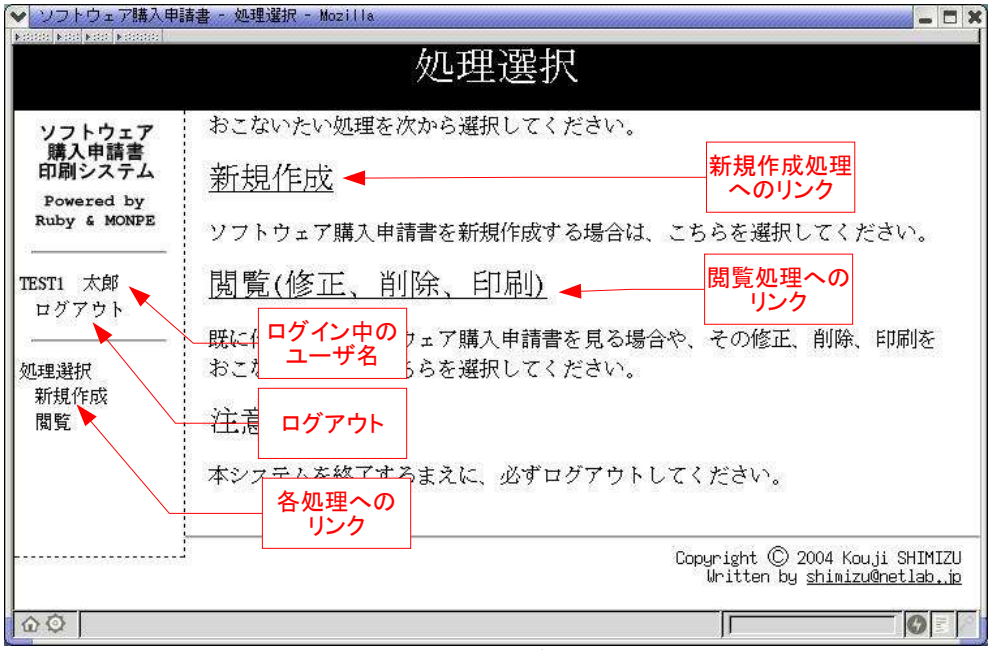

図 3.1 処理選択画面

新規作成
　必要な情報をデータベースに登録し、ソフトウェア購入申請書を新規作成できます。

閲覧
　データベースに登録してソフトウェア購入申請書を見ることができます。また、登録したものを修正、削除、印刷といったこともおこなえます。

ログアウト
　本システムを終了するときは、ログアウトを実行してください。

4.新規作成

本システムを使ってソフトウェア購入申請書を印刷するには、まず、購入するソフトウェア名、購入理由や金額などの必要な情報をデータベースに登録する必要があります。そして、その情報を使ってソフトウェア購入申請書を印刷します。

ここでは、新しいソフトウェア購入書を作成し、それを印刷する方法について説明します。

Step1.新規作成の選択

処理選択画面から新規作成を選択します。

Step2.必要な項目の入力

図 4.2 のような入力画面が表示されますので、必要な情報を入力してください。

入力後は、作成ボタンを押してください。

このうち、入力項目名の横に「*」が表示されているものは、入力しなければいけない項目です。また、明細行は、最低一つは入力する必要があります。

合計金額などを入力する項目がないのは、システムが自動で計算するので、入力する必要がないからです。

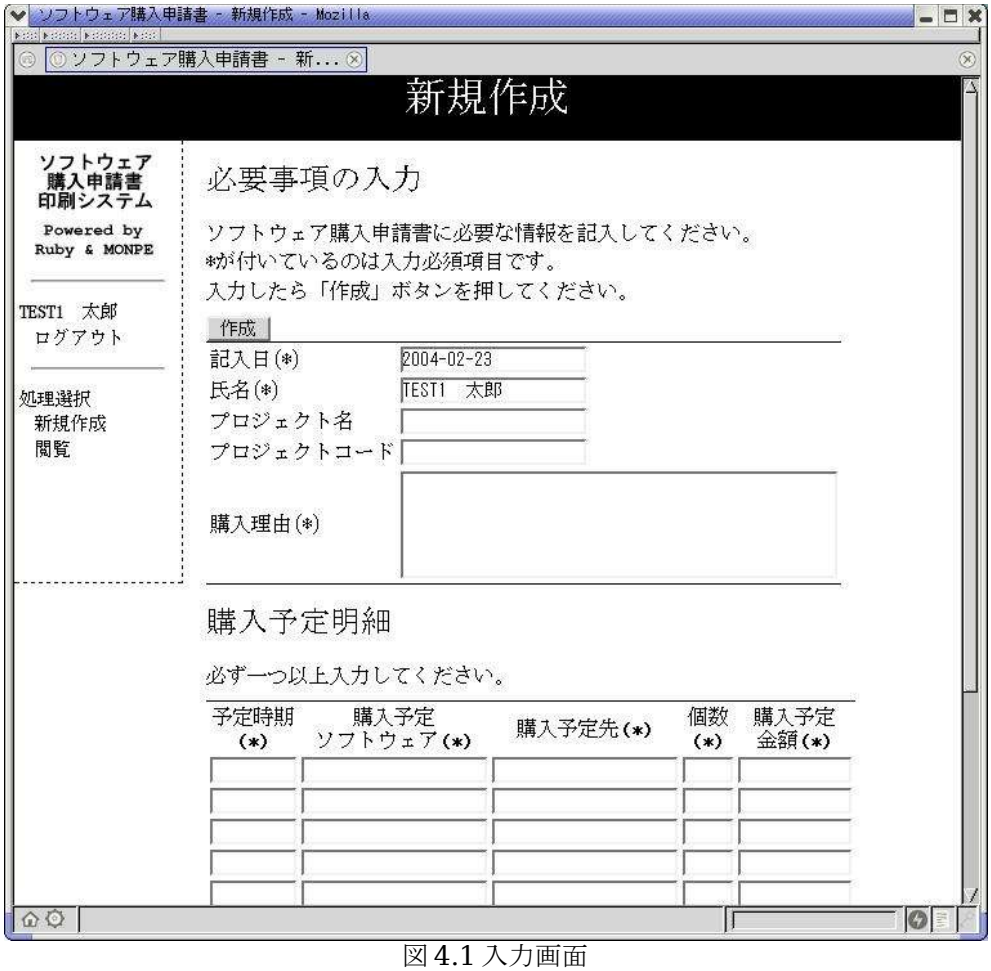

図 4.1 入力画面

Step3.入力内容の確認

記入日の入力欄に氏名を入力してしまうなどの入力間違いがなければ、図 4.2 のような入力確認画面が表示されます。

入力した内容で作成してよければ、作成実行ボタンを押してください。

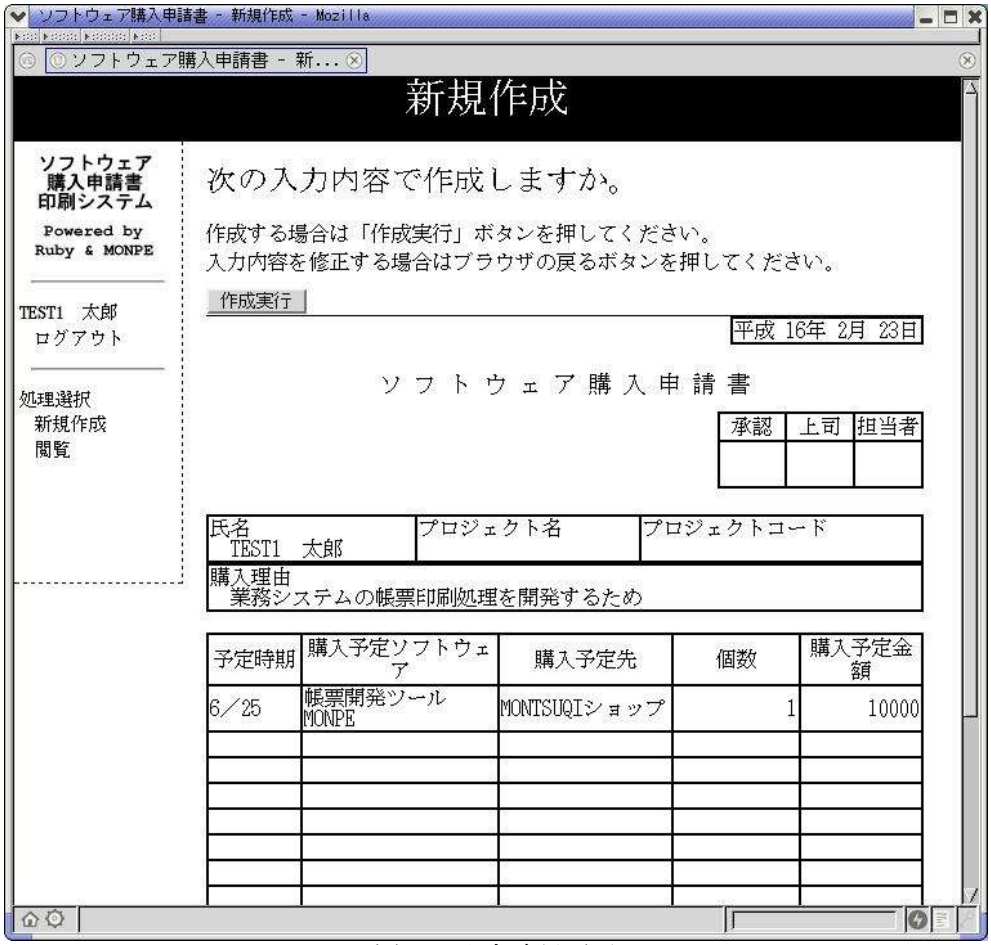

図 4.2 入力確認画面

もし、入力に間違いがあった場合は、図 4.3 のような入力エラー画面が表示されます。入力に間違いがあった項目には色がついていますので、その情報をもとに、再度、必要な情報を入力してください。入力後は、作成ボタンを押してください。

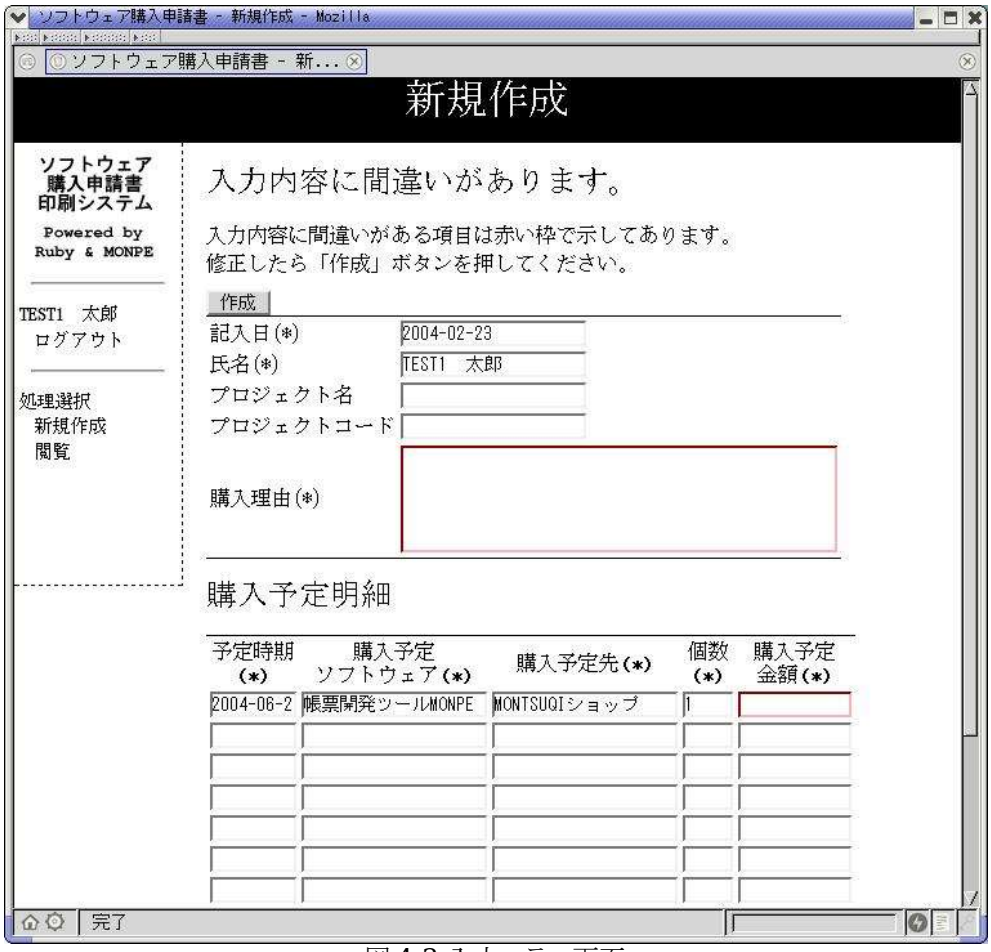

図 4.3 入力エラー画面

Step4.作成完了

入力した情報がデータベースに作成されると図 4.4 のような作成完了画面が表示されます。

すぐに、入力した情報でソフトウェア購入申請書を印刷するには、プリンタ選択リストから希望するプリンタを選択し、印刷ボタンを押してください。

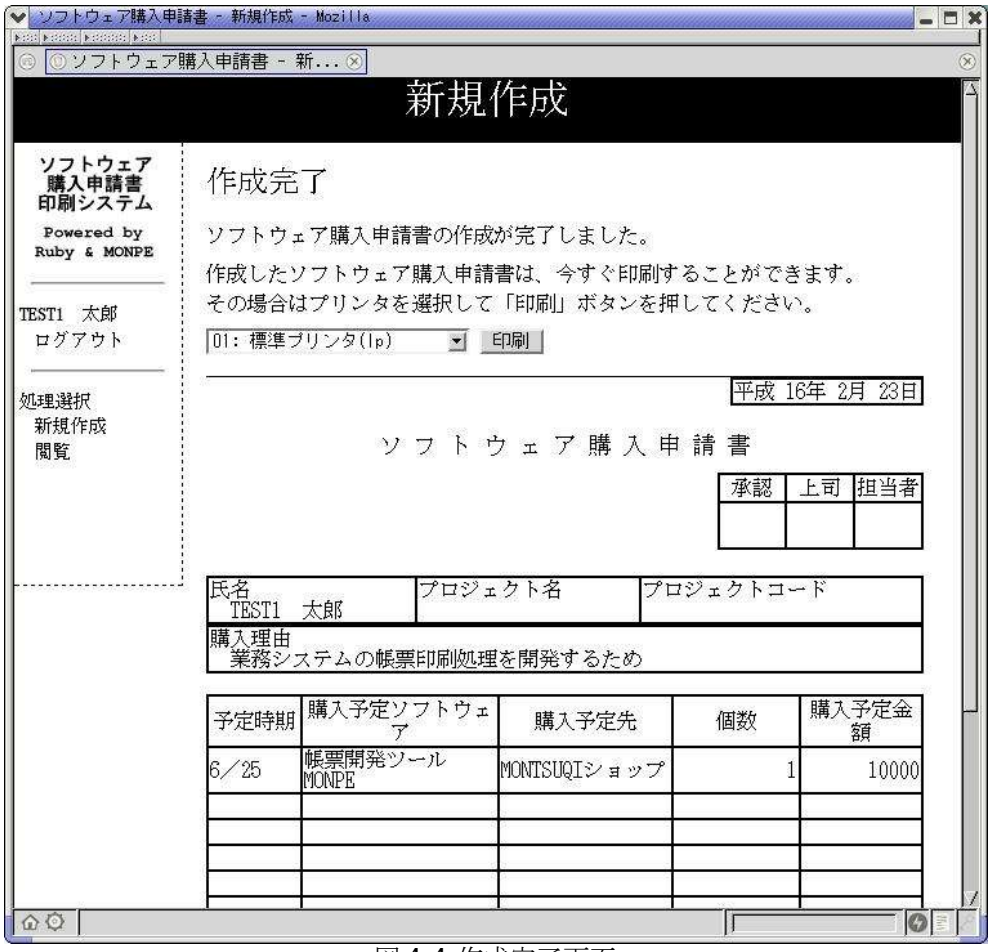

図 4.4 作成完了画面

Step5.印刷

印刷が完了すると、図 4.5 のような印刷完了画面が数秒表示され、その後、自動的に処理選択画面に移り変わります。

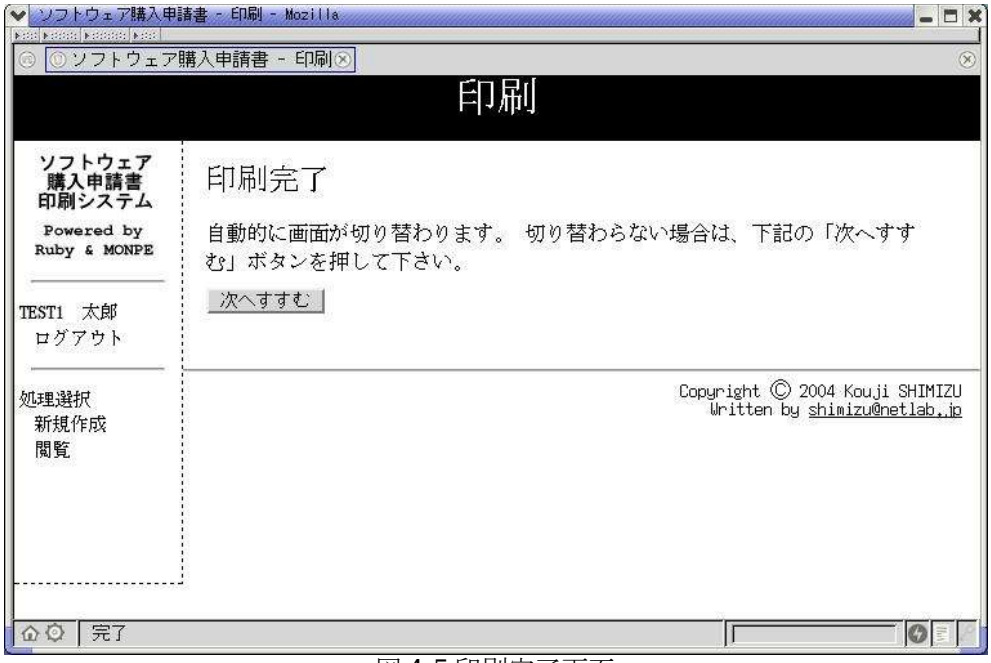

図 4.5 印刷完了画面

5.閲覧

データベースに登録したソフトウェア購入申請書の情報は、いつでも見ることができます。

ここでは、データベースに登録したソフトウェア購入申請書の情報を見ることについて説明します。

Step1.閲覧の選択

処理選択画面から閲覧を選択します。

Step2.閲覧するソフトウェア購入申請書の選択

閲覧を選択すると、図 5.1 のような概要一覧画面が表示されます。この画面には、データベースに登録されているソフトウェア購入申請書の概要が一覧表示されています。詳しい情報が見たいものの閲覧ボタンを押してください。

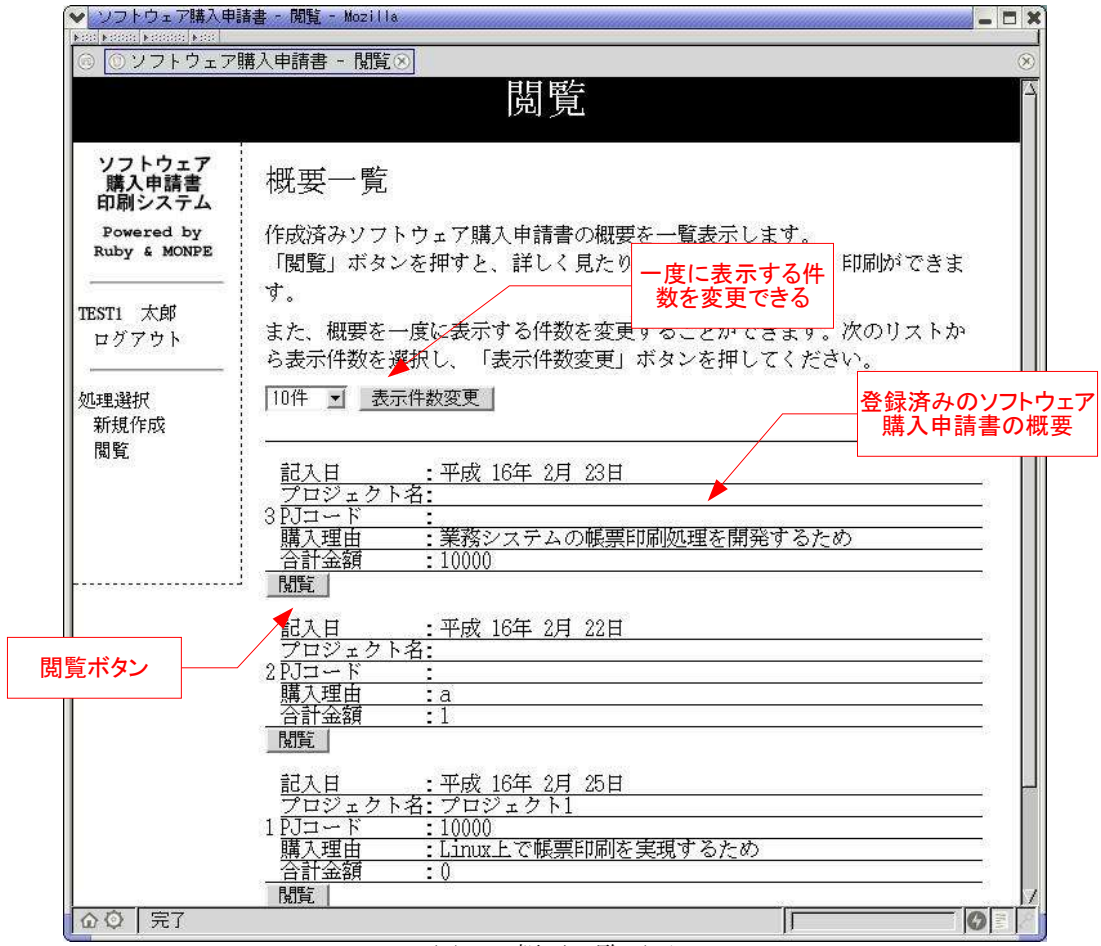

図 5.1 概要一覧画面

一度に表示する件数は、表示件数リストから希望の件数を選択後、変更ボタンを押すことで変更できます。

Step3.ソフトウェア購入申請書の閲覧

閲覧ボタンを押すと、図 5.2 のような閲覧画面に、ソフトウェア購入申請書の詳しい情報が表示されます。

ここで、修正、削除、印刷ボタンを押すと、それぞれ表示しているソフトウェア購入申請書の修正、削除、印刷ができます。それぞれの処理については別の章で説明します。

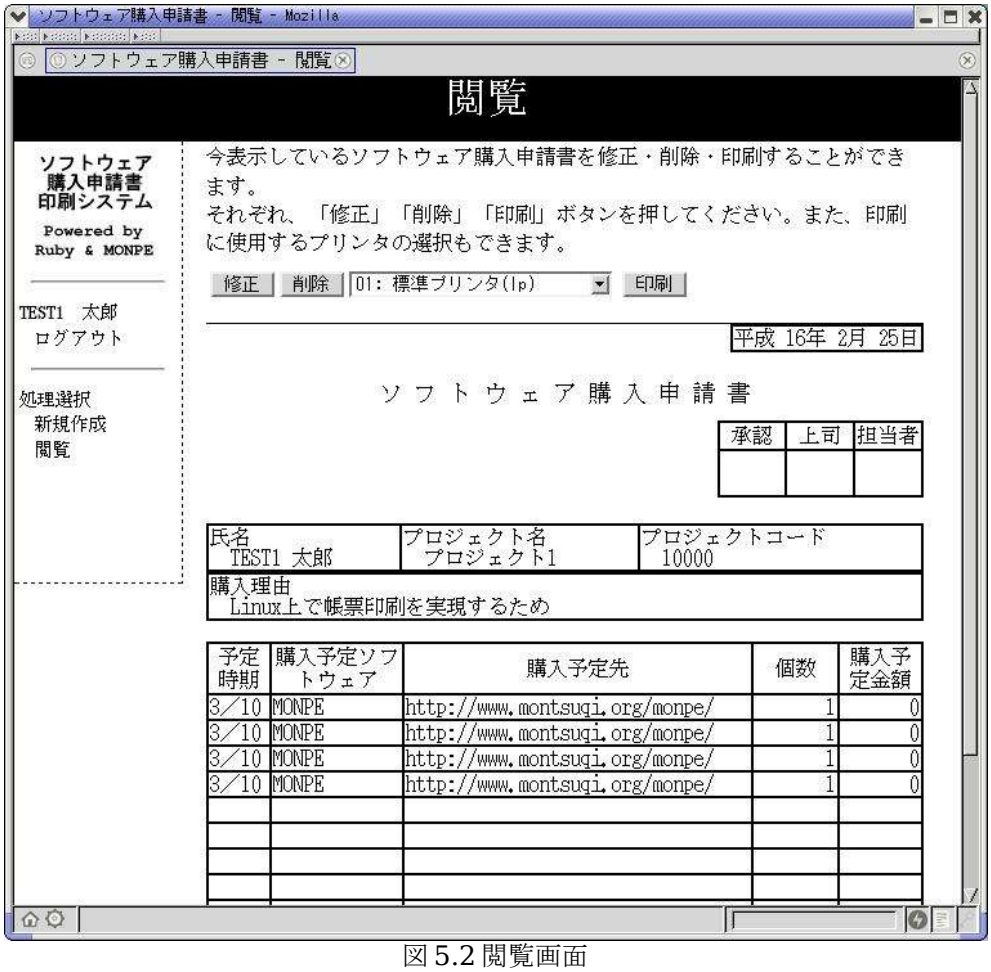

図 5.2 閲覧画面

6.修正

閲覧画面の修正ボタンを押すことで、データベースに登録してあるソフトウェア購入申請書の情報を修正することができます。

ここでは、その方法について説明します。

Step1.修正内容の入力

閲覧画面の修正ボタンを押すと、図 6.1 のような修正画面が表示されます。新規作成のときとは違い、いくつかの入力項目は、修正対象のソフトウェア購入申請書の情報が入力してあります。項目を修正したら、修正ボタンを押してください。

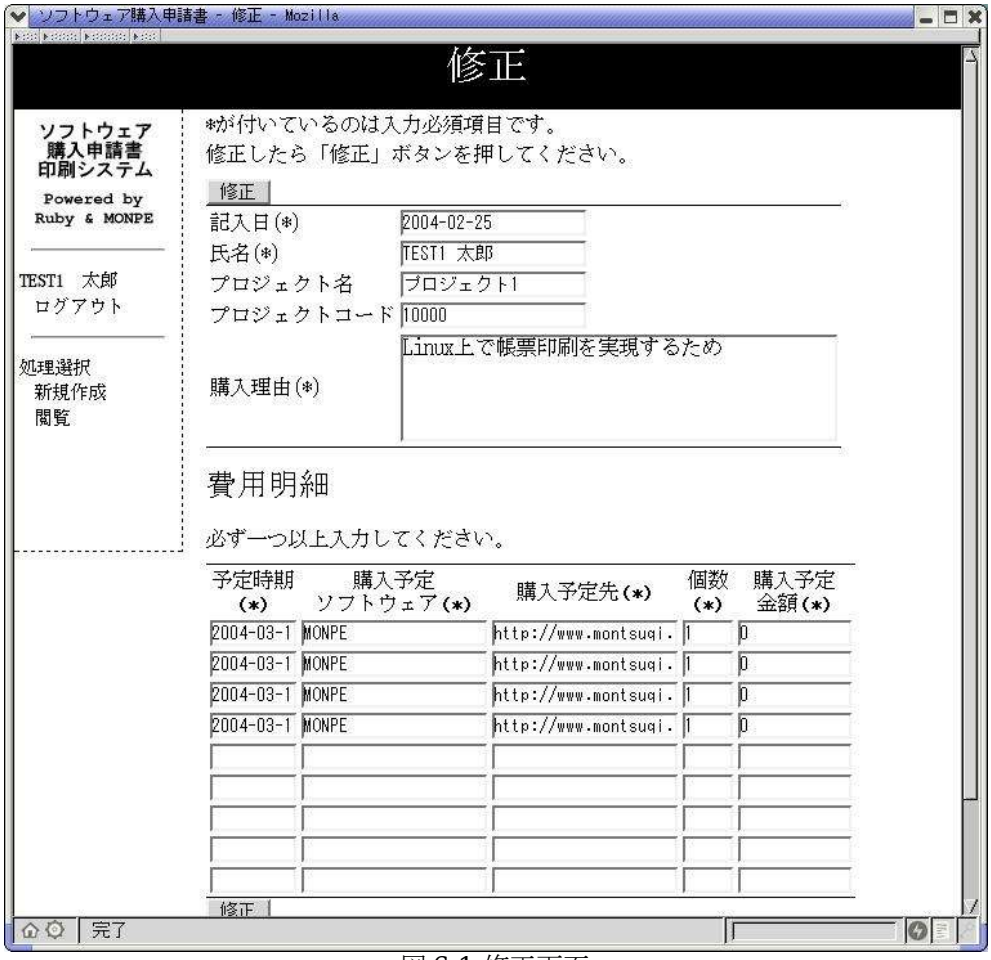

図 6.1 修正画面

Step2.修正内容の確認

入力間違いがなければ、図 6.2 のような修正確認画面が表示されます。
内容に間違いがなく、修正してよければ、修正実行ボタンを押してください。

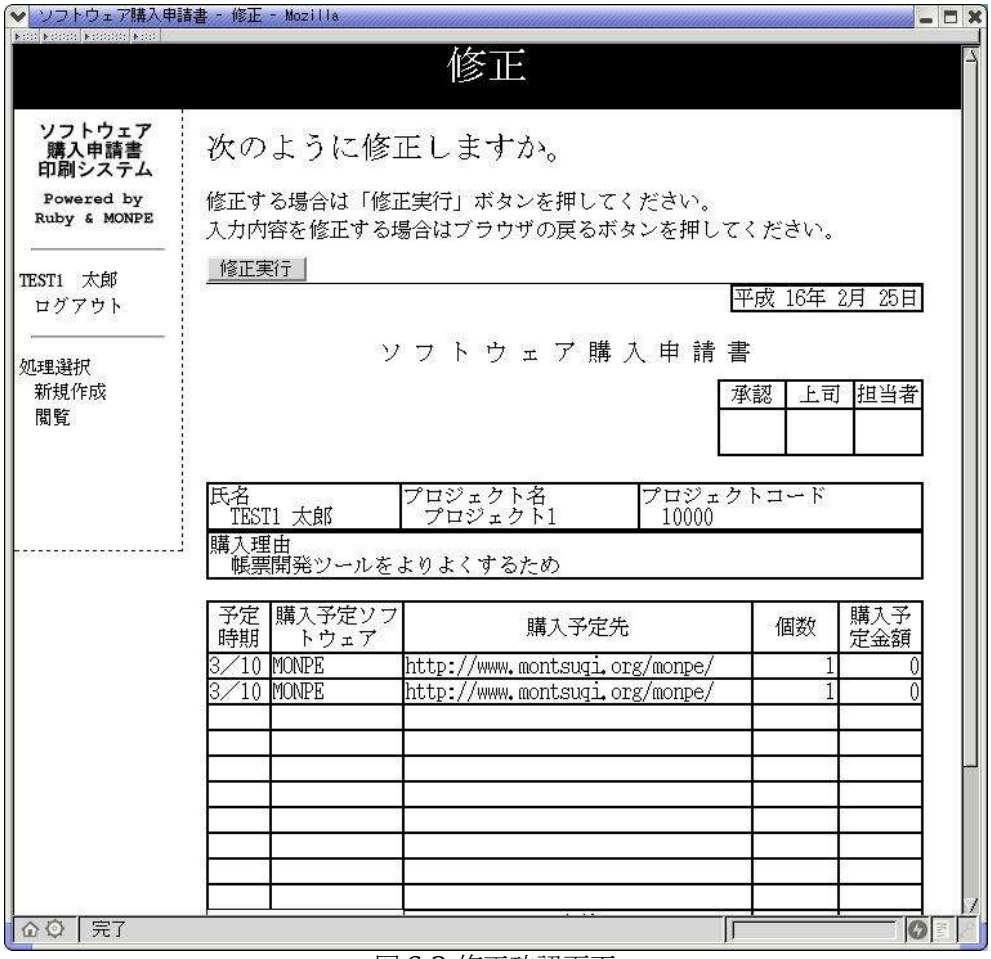

図 6.2 修正確認画面

もし、入力に間違いがあった場合は、図 6.3 のような修正エラー画面が表示されます。入力に間違いがあった項目には色がついていますので、その情報をもとに、再度、必要な情報を入力してください。入力後は、修正ボタンを押してください。

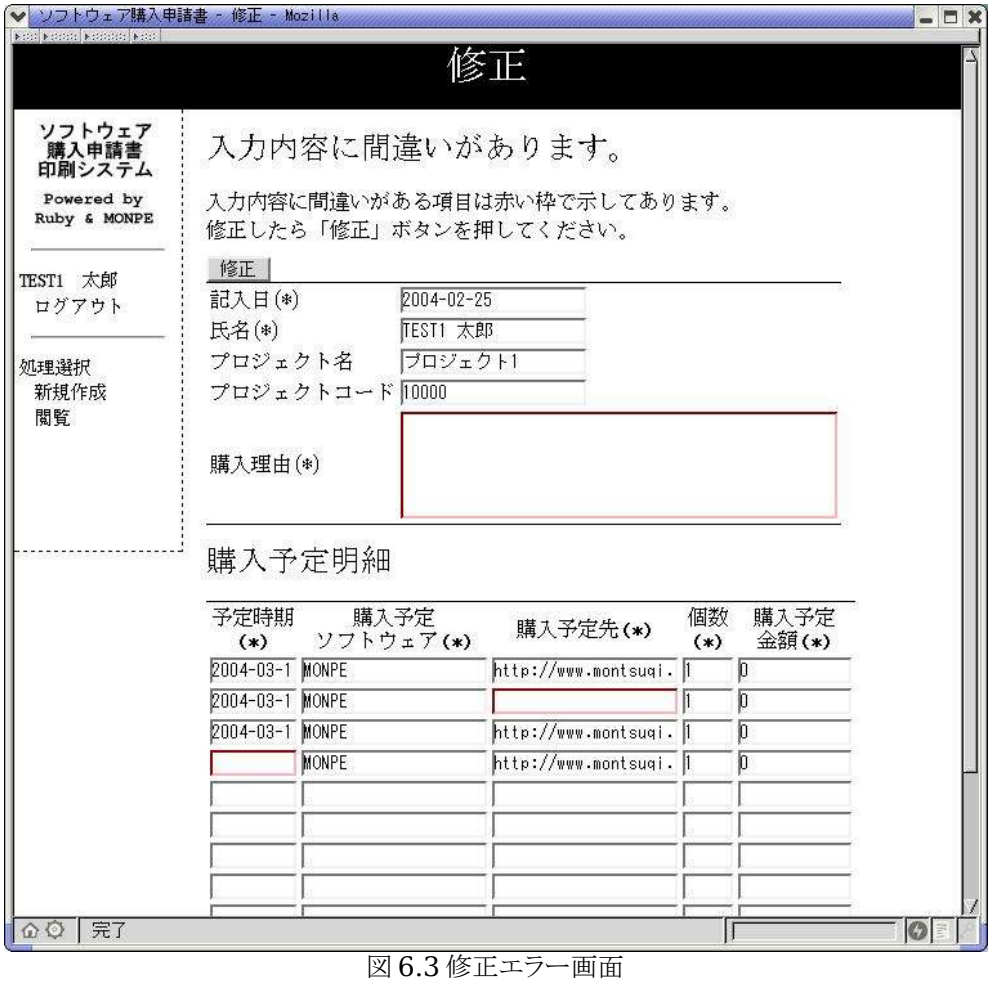

図 6.3 修正エラー画面

Step3.修正完了

無事、ソフトウェア購入申請書が更新されると図 6.4 のような修正完了画面が表示されます。

すぐに、ソフトウェア購入申請書を印刷するには、プリンタ選択リストから希望するプリンタを選択し、印刷ボタンを押してください。

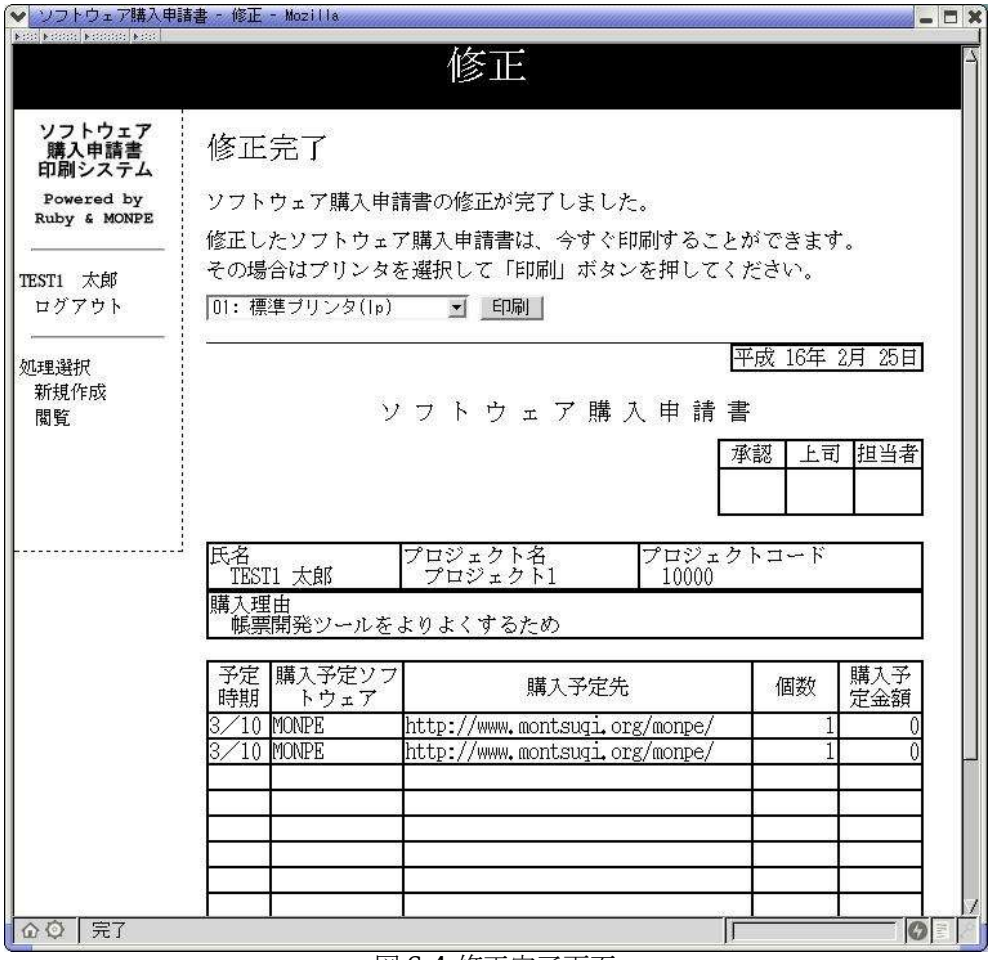

図 6.4 修正完了画面

Step4.印刷

印刷が完了すると、印刷完了画面が数秒表示され、その後、処理選択画面に自動的に移り変わります。

7.削除

閲覧画面の削除ボタンを押すことで、データベースに登録してあるソフトウェア購入申請書の情報を削除することができます。

ここでは、その方法について説明します。

Step1.削除の確認

閲覧画面の削除ボタンを押すと、図 7.1 のような削除確認画面が表示されます。本当に削除する場合は、削除実行ボタンを押してください。

削除してしまった情報を復元することはできませんので、ご注意ください。

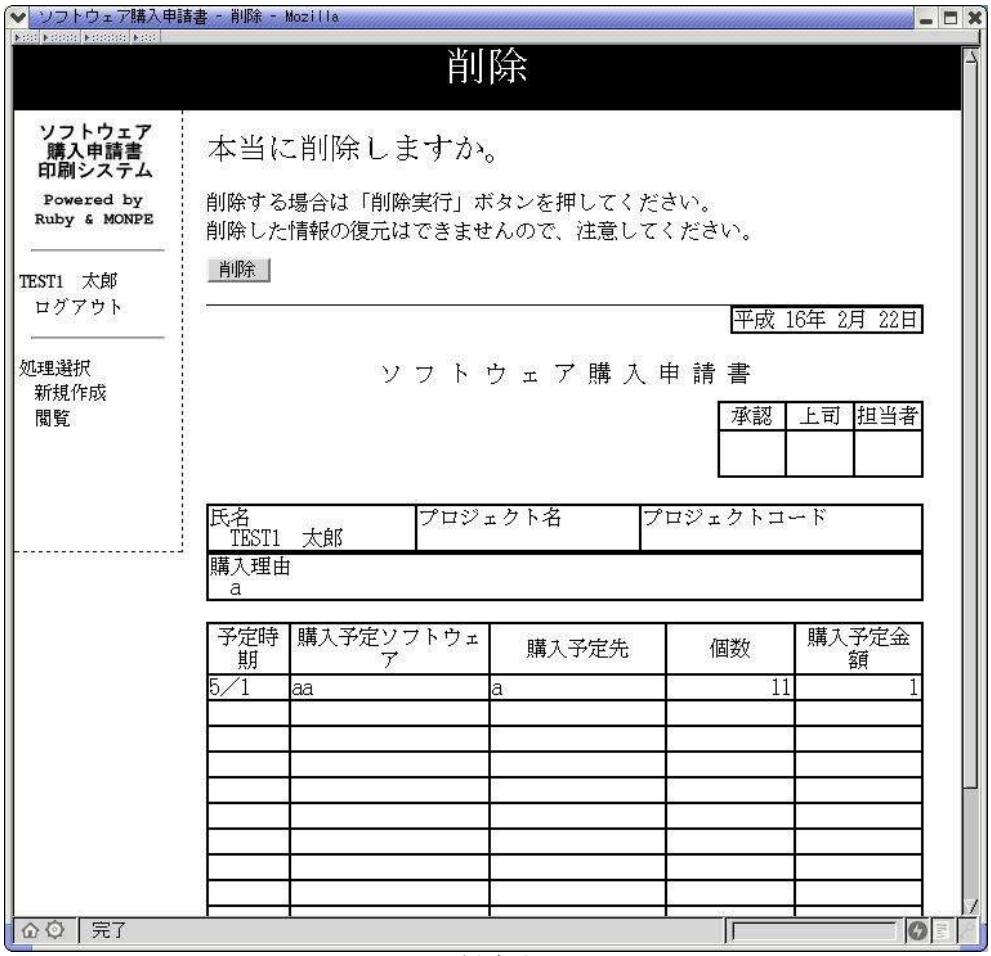

図 7.1 削除確認画面

Step2.削除完了

　データベースから情報の削除が完了すると、図 7.2 のような削除完了画面が数秒表示され、その後、自動的に処理選択画面に移り変わります。

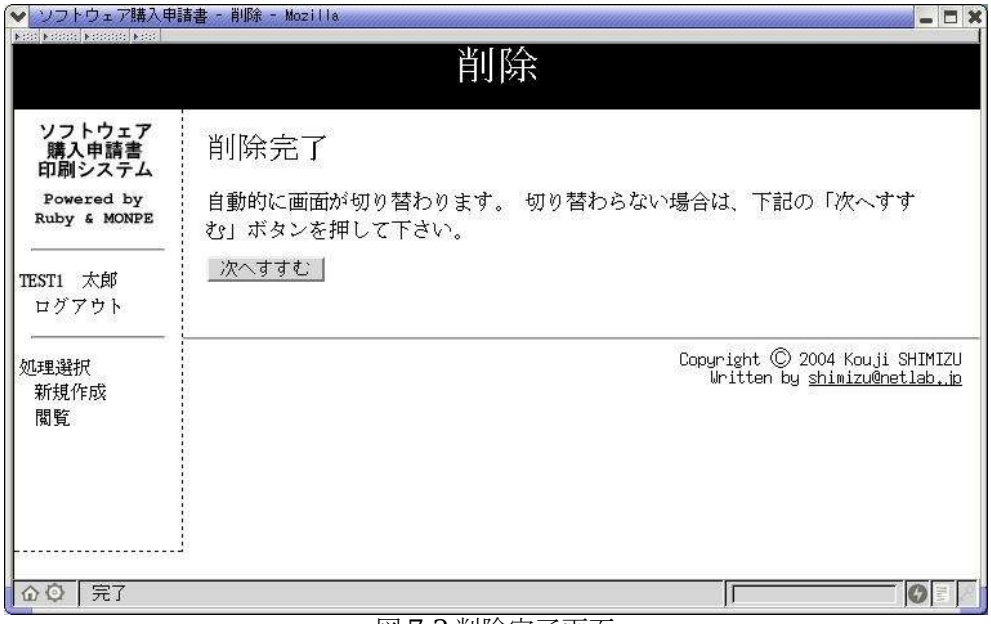

図 7.2 削除完了画面

8.印刷

閲覧画面の印刷ボタンを押すことで、データベースに登録してあるソフトウェア購入申請書を印刷することができます。

ここでは、その方法について説明します。

Step1.印刷の実行

閲覧画面のプリンタ選択リストから希望するプリンタを選択後、印刷ボタンを押すと、図8.1 のような印刷完了画面が数秒表示され、その後、自動的に処理選択画面に移り変わります。

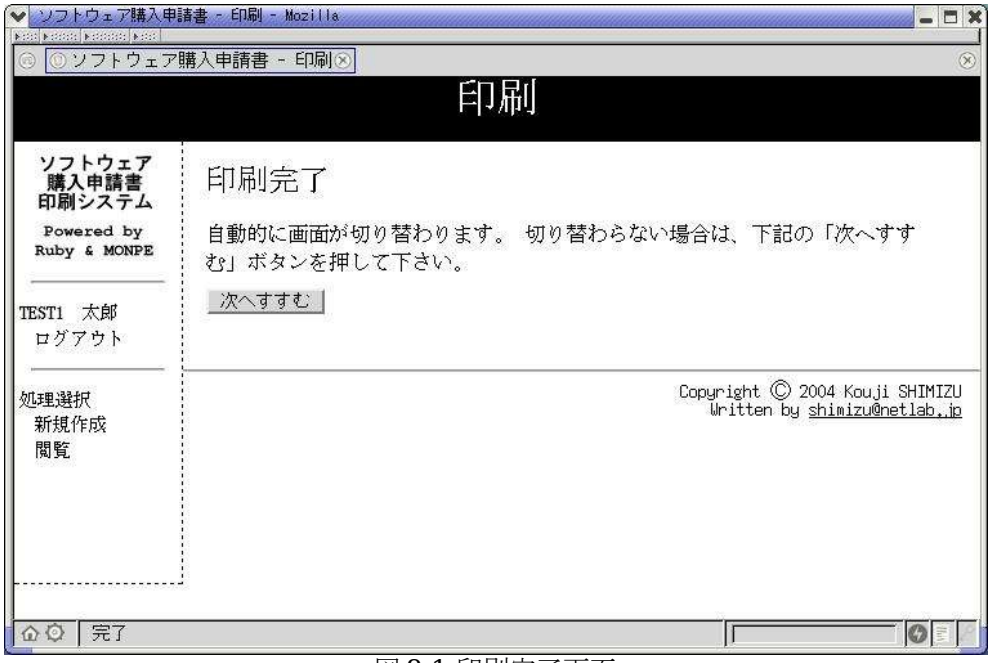

図 8.1 印刷完了画面

9.ログアウト

本システムを終了するときは、ログアウトを実行してください。

ここでは、ログアウトをおない、本システムを終了する方法方法について説明します。

Step1.ログアウトの選択

ログアウトするには、画面左にあるメニューから、ログアウトを選択してください。すると、図9.1 のようなログアウト中画面が数秒表示され、その後、自動的にログイン画面に移り変わります。

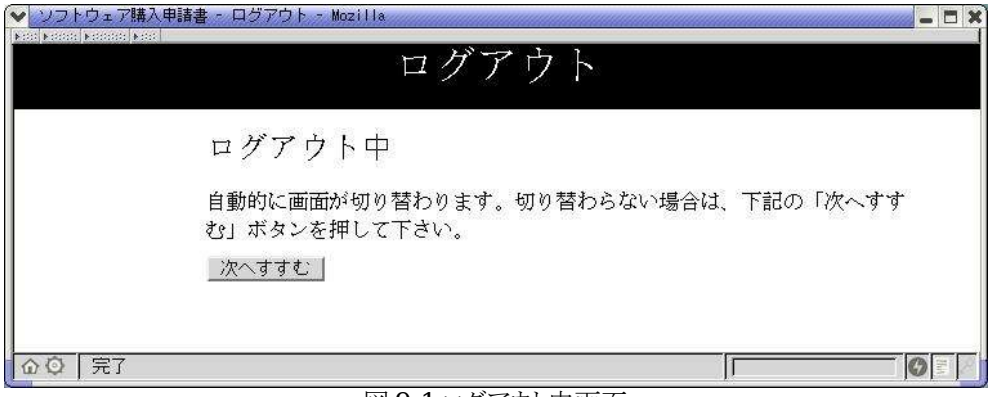

図 9.1 ログアウト中画面

Step2.本システムの終了

ブラウザの閉じてください。これで、本システムが終了しました。

付録 A.ディレクトリ構成

ここでは、本システムのディレクトリ構成について説明する。

- softshinsei
  - application.css：本システムで使用する HTML のスタイルシート
  - config-test.rb：単体テスト用システム設定ファイル
  - config.rb：システム設定ファイル
  - database：データベース操作する Ruby スクリプトを格納
  - delete.cgi：削除完了画面
  - delete_check.cgi：削除確認画面
  - html-part：HTML 動的に生成する Ruby スクリプトを格納
  - input.cgi：入力画面
  - input_check.cgi：入力確認画面、入力エラー画面
  - insert.cgi：作成完了画面
  - logic：業務処理をおこなう Ruby スクリプトを格納
  - login.cgi：ログイン画面
  - login_now.cgi：ログイン中画面
  - logout.cgi：ログアウト画面
  - modify.cgi：修正画面
  - modify_check.cgi：修正確認画面、修正エラー画面
  - printout.cgi：印刷完了画面
  - proc_select.cgi：処理選択画面
  - report：MONPE で設計したソフトウェア購入申請書を格納
  - report.png：画面左上の画像
  - session：セッション情報を格納
  - sql：システムに必要なデータベースを構築する SQL を格納
  - summary.cgi：概要一覧画面
  - tests：単体テストをおこなう Ruby スクリプトを格納
  - tests-run.sh：単体テストをおこなうシェルスクリプト
  - update.cgi：修正完了画面
  - utils.rb：よく使用する関数を定義した Ruby スクリプト
  - validater.rb：入力内容をチェックする Ruby スクリプト
  - view.cgi：閲覧画面

付録 B.ユーザの追加・修正・削除 SQL

ここでは、ユーザの追加・修正・削除をおこなうSQLについて説明します。

- ユーザの追加

  INSERT INTO employee (emp_no, password, last_name_j, first_name_j) VALUES(社員番号, 'パスワード', '名字', '名前');

  例) 社員番号「10」、パスワード「hoge」、名字「ホゲ」、名前「太郎」のユーザを追加する。
  INSERT INTO employee (emp_no, password, last_name_j, first_name_j) VALUES(10, 'hoge', 'ホゲ', '太郎');

- ユーザの修正

  UPDATE employee SET password = 'パスワード', last_name_j = '名字', first_name_j = '名前'
  WHERE emp_no = 社員番号;

  例1) 社員番号10のパスワードを「test」に変更する。
  UPDATE employee SET password = 'test' WHERE emp_no = 10;

  例2) 社員番号10の名字を「テスト」に変更する。
  UPDATE employee SET last_name_j = 'テスト' WHERE emp_no = 10;

  例3) 社員番号10の名前を「二郎」に変更する。
  UPDATE employee SET first_name_j = '二郎' WHERE emp_no = 10;

- ユーザの削除

  DELETE FROM employee WHERE emp_no = 社員番号;

  例) 社員番号10のユーザを削除する。
  DELETE FROM employee WHERE emp_no = 10;

参照

- sql/testdata-employee.sql

付録 C.プリンタの追加・修正・削除 SQL

ここでは、本システムで使用するプリンタの追加・修正・削除をおこなうSQL について説明します。

- プリンタの追加

  INSERT INTO printer (printer_no, lpr, name, description) VALUES (プリンタ番号, 'lpr エントリ名(※)', 'プリンタ名', 'プリンタの概要');
  ※lpr エントリ名には、/etc/printcap に設定したプリンタ名を指定します。

  例) プリンタ番号「10」、lpr エントリ名「rlp」、プリンタ名「リモートプリンタ」、プリンタの概要「リモート(192.168.0.133)プリンタ」を追加する。
  INSERT INTO printer (printer_no, lpr, name, description) VALUES (10, 'rlp', 'リモートプリンタ', 'リモートプリンタ(192.168.0.133)');

- プリンタの修正

  UPDATE printer SET lpr = 'lpr エントリ名', name = 'プリンタ名', description = 'プリンタの概要' WHERE printer_no = プリンタ番号;

  例 1) プリンタ番号 10 の lpr エントリ名を「rlp2」に変更
  UPDATE printer SET lpr = 'rlp2' WHERE printer_no = 10;

  例 2) プリンタ番号 10 のプリンタ名を「リモートプリンタ 2」、プリンタの概要を「リモートプリンタ 2(192.168.0.200)」に変更
  UPDATE printer SET name = 'リモートプリンタ 2', 'リモートプリンタ 2(192.168.0.200)' WHERE printer_no = 10;

- プリンタの削除

  DELETE FROM printer WHERE printer_no = プリンタ番号;

  例) プリンタ番号 10 のプリンタを削除する。
  DELETE FROM printer WHERE printer_no = 10;

参照

- sql/testdata-printer.sql